Chapter IX

A Comparison of the FOOM and OPM Methodologies for User Comprehension of Analysis Specifications

Judith Kabeli, Ben-Gurion University of the Negev, Israel
Peretz Shoval, Ben-Gurion University of the Negev, Israel

ABSTRACT

FOOM (Functional and Object-Oriented Methodology) and OPM (Object- Processes Methodology) are methodologies used for analyzing and designing information systems. Both integrate functional and object-oriented approaches, but differ in that the analysis specification of FOOM utilizes OO-DFDs (Data Flow Diagrams with object classes that replace traditional data-stores) and a class diagram, while OPM defines a new notational model for specifying the system’s structural and procedural requirements, which combines processes and classes in a unified diagrammatic notation. In this study, we compare FOOM and OPM from the point of view of both user comprehension of analysis specifications and user preference of specifications. The comparison is based on a controlled experiment that measured: (a) comprehension of the analysis specifications,
which includes both structural and behavioral aspects of the system; (b) the
time it takes to complete the task of specification comprehension; and (c)
the user’s preference of models. Our results show that FOOM performed
better than OPM and that the users preferred FOOM to OPM.

INTRODUCTION

FOOM is a methodology used for analyzing and designing information
systems. It combines two essential software-engineering paradigms: the func-
tional/data approach (or process-oriented) and the object-oriented (OO) ap-
proach (Shoval & Kabeli, 2001). FOOM utilizes known methods and techniques
such as DFDs (Data Flow Diagrams), and provides simple visual modeling and
notations. It covers the structural and the behavioral aspects of a system through
the analysis and design phases, and provides a natural and smooth transition from
one stage to the other. Since users express their information needs in terms of
both functions and data structure — rather than in terms of object classes and
their behavior — it seems to us that an appropriate method to carry out the
analysis task is to combine functional and data analysis. For designing an
information system, however, OO design should be applied since it has been
shown to be a better approach to implement software.

The Object-Processes Methodology (OPM) is used for analysis and design
of information systems. It combines process and object approaches, and
provides a unified notation for the structural and behavior aspects of a system
(Dori, 2001). OPM utilizes a single graphic tool, the Object-Process Diagram
(OPD), in all of its development phases. However, since OPD defines a new
notation that combines DFD and OO diagrams, it incorporates many symbols and
rules. In our opinion, such diagrams are not very easy to construct and
comprehend for large-scale systems. In our mind, reality should be modeled with
simple notations that are easy to comprehend and utilize. A single hybrid notation
like OPM must be very rich in order to elicit all points of view, thus leading to a
complex, perhaps misunderstood model of reality. On the other hand, multiplicity
of models and corresponding diagramming tools, as in UML (Unified Modeling
Language), may also be too complicated. Too many diagram types may hamper
coherent understanding and lead to the production of erroneous models and systems.

Methodologies can be evaluated and compared for various dimensions, e.g.,
quality of the analysis and design products, comprehensibility, learning ability,
ease of use, preference by users or professional developers, and others. In this
chapter, we present an experimental comparison of FOOM and OPM method-
ologies from the point of view of end-user comprehension of specifications, time
to complete tasks of comprehension, and user preference of models to determine
which methodology is better comprehensible. OPM was selected to be com-
pared with FOOM since it also combines and integrates the two essential
Related Content

Method Chunks to Federate Development Processes
www.irma-international.org/chapter/method-chunks-federate-development-processes/28436/

Goal-Oriented Requirement Engineering for XML Document Warehouses
www.irma-international.org/chapter/goal-oriented-requirement-engineering-xml/7932/

www.irma-international.org/article/understanding-business-domain-models/49724/

An ER Based Methodology for Modeling User Views and Detecting Derived Relationships
Dinesh Batra and Jeffrey A. Hoffer (1994). Journal of Database Management (pp. 3-17).
www.irma-international.org/article/based-methodology-modeling-user-views/51128/

Organizational Memory Management: Technological and Research Issues
www.irma-international.org/article/organizational-memory-management/3349/